UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/810,167	03/19/2001	Gerald George Kiernan	ARC920010026US1	9252

26381 7590 04/19/2004

LACASSE & ASSOCIATES, LLC
1725 DUKE STREET
SUITE 650
ALEXANDRIA, VA 22314

EXAMINER	
PHAM, HUNG Q	

ART UNIT	PAPER NUMBER
2172	8

DATE MAILED: 04/19/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/810,167	KIERNAN ET AL.	
	Examiner	Art Unit	
	HUNG Q PHAM	2172	

-- The MAILING DATE of this c mmunication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 March 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-46* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3,4,10-13,18-22,24,25,31-34 and 39-45* is/are rejected.

7)☒ Claim(s) *2,5-9,14-17,23,26-30,35-38 and 46* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

1. Applicant's arguments filed 03/04/2004 have been fully considered but they are not persuasive.

As argued by applicants as in page 16, lines 13-23:

While the disclosed method (Fernandez) provides for an executable query produced by a query composer module composition of an RXL view query and an XML-QL user query ... The present invention discloses a default XML view upon which more complex views are derived - the tagger tree graph corresponding to a particular XML view is derived at run-time from an XML query directly. In this manner, any hierarchical structure specified by an initial XML query is supported. The method disclosed by Fernandez does not mention how relational data is converted to XML, nor does it mention the module facilitating its conversion is generated at runtime.

Page 17, lines 4-6:

Fernandez does not teach nor suggest a system for tagging or structuring the results of an SQL query of XML data stored in a relational database.

Page 17, lines 12-13:

The disclosed method (Fernandez) does not provide for arbitrarily joining XML user queries or resultant XML data.

Page 17, lines 18-20:

The method of the present invention provides for arbitrary and dynamic views since a tagger tree graph is generated at run-time, and XML results are hierarchically tagged as they are produced.

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies above are not recited in the rejected claim(s). Although the claims are interpreted in

light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

3. Claims 1, 3-4, 10-13, 18-22, 24-25, 31-34 and 39-45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fernandez et al. [USP 6,604,100].

Regarding to claims 1, 22 and 44-45, Fernandez teaches a method, a system, a computer program product for converting relational data to XML. The Fernandez

method utilizes SilkRoute to use an existing XML query language, XML-QL, against a relational database (Col. 3, Lines 10-24). As shown in FIG. 1, SilkRoute 100 serves as middleware between a relational database server 110, and an application 120. To access the data, the application 120, can formulate a user query in XML-QL over the virtual view and send it to SilkRoute 100. Together, the view query and the user query can be passed to the query composer module 102. The query-composer module computes the composition and produces a new view query, RXL query, called the executable query. The answer to the executable query typically includes only a small fragment of the database, e.g., one data item, a small set of data items, or an aggregate value. The result of SilkRoute 100 is an XML document, as specified by the user query (Col. 4, Line 64-Col. 5, Line 25). As shown in FIG. 2, the query is composed by pattern matcher 140 by evaluating user queries (U) on view query (V) templates (Col. 12, Lines 32-45). The view query V may be represented by a data structure called a view tree (Col. 12, Lines 60-61) as ***a tagger tree graph***. A view tree Node is composed of a tag, a rule, and a list of children nodes. A Rule is composed of a Skolem term as its head, and a conjunctive list of conditions in its body. A Condition is a table expression, a filter expression, or the disjunction of two lists of conjuncts (Col. 21, Lines 43-48). As seen, each node of a view tree comprises a rule including a condition such as a table expression, a filter expression, or the disjunction of two lists of conjuncts as ***a tagger operator***, and a list of children nodes as ***a parse tree***. In other words, the Fernandez technique as discussed indicates the step of ***generating a tagger tree graph from said XML query, each node of said tagger tree graph comprising a tagger operator, each tagger operator***

having a parse tree associated therewith. Fernandez does not explicitly teach the step of ***calling each tagger operator in accordance with a structure of said tagger tree graph, and evaluating said parse trees associated with each called tagger operator to tag results of said XML query over said relational database***. However, as illustrated by Fernandez, a reseller can retrieve all products with a sale price less than half of the retail price using the XML-QL user query (U) below:

```
1.  construct
2.  <results> {
3.      where <supplier>
4.              <company>$company</company>
5.          <product>
6.            <name>$name</name>
7.            <retail>$retail</retail>
8.            <sale>$sale</sale>
9.          </product>
10.       </supplier> in "http://acme.com/products.xml",
11.       $sale < 0.5 * $retail
12.   construct
13.         <result ID=Result($company)>
14.           <supplier>$company</supplier>
15.           <name>$name</name>
16.         </result>
17. } </results>
```

The query composer module 102 takes the user query and the RXL view query and generates a new RXL query. The RXL view query (V) is constructed as below

```
1.  construct
2.  <supplier ID=Supp( )>
3.      <company ID=Comp ( )>"Acme Clothing"</company>
4.      {
5.      from Clothing $c
6.      where $c.category = "outerwear"
7.      construct
8.           <product ID=Prod($c.pid)>
9.           <name ID=Name($c.pid,$c.item)>$c.item</name>
10.          <category ID=
             Cat($c.pid,$c.category)>$c.category</category>
11.          <descriptionID=
          Desc($c.pid,$c.description)>$c.description</description>
12.          <retail
```

```
          ID=Retail($c.pid,$c.price)>$c.price</retail>
13.        { from SalePrice $s
14.           where $s.pid = $c.pid
15.           construct
16.           <sale
          ID=Sale($c.pid,$s.pid,$s.price)>$s.price</retail>
17.        }
18.        { from Problems $p
19.           where $p.pid = $c.pid
20.           construct
21.           <report code=$p.code ID=
              Prob($c.pid,$p.pid,$p.code,$p.comments)>
22.              $p.comments
23.           </report>
24.        }
25.        </product>
26.    }
27. </supplier>
```

And the composed query, RXL query (C) is shown below.

```
construct
    <results>
{ from Clothing $c, SalePrice $s
    where $c.category = "outerwear",
        $c.pid = $s.pid,
        $s.price < 0.5 * $c.retail
    construct
        <result ID=Result("Acme Clothing")>
        <supplier>"Acme Clothing"</supplier>
        <name ID=Name($c.pid, $c.item)>$c.item</name>
    </result>
}
</results>
```

The translator 104 takes an RXL query and decomposes it into one or more SQL queries and an XML template. The SQL queries are executed by the RDBMS 110, and their flat results, streams of tuples, are converted into XML by the XML generator 106 (Col. 9, Lines 61-65 and Col. 10, Lines 15-20).

```
          <results>
             <result ID=Result("Acme Clothing")>
             <supplier>"Acme Clothing"</supplier>
             <name ID=Name($pid, $item)>$item</name>
          </result>
```

RXL enforces semantic constraints that guarantee a view always defines a tree (Col. 7, Lines 8-9), and the created RXL query corresponds to a list of RXL blocks (Col. 22, Lines 53-56), which is created as below (Col. 23, Lines 20-35):

```
1.  // Return new RXL block for potential solution in r_i
2.  fun oneSolution(Env X_env VarMap S, XMLQL X_block,
    Env r_i) : [RXL] {
3.   R_conditions = new [ ]
4.   // For each XML-QL variable X_v in X_block
5.   foreach X_v in getVariables(X_block) {
6.   // Get view-tree node bound to X_v
7.   R_node = project(r_i, X_v);
8.   // Get rule associated with view-tree node
9.   (R_tag, R_rule R_children) = R_node
```

As seen, XML generator ***tags*** the ***results of an XML query over a relational database***, and in order to have the returned XML document, the view tree node bound to the query is obtained, the rules associated with view tree node include R_tag as ***tagger operator in accordance with*** the view tree node or ***a structure of tagger tree graph***, R_children as ***parse tree associated with*** the corresponding ***tagger operator*** are called, and obviously, R_children must be ***evaluated*** to guarantee a view always defines a tree. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Fernandez method by including the step of evaluating parse tree in order to return the result to an XML query.

Regarding to claims 3 and 24, Fernandez teaches all the claimed subject matters as discussed in claims 1 and 22, Fernandez further discloses ***tagger operators comprise any of a tagger input operator, a tagger scalar operator or a tagger aggregate operator*** (Col. 20, Lines 44-49).

Regarding to claims 4 and 25, Fernandez teaches all the claimed subject matters as discussed in claims 1 and 22, Fernandez further discloses ***tagger graph includes a tagger input operator for each level in a result XML tree of said XML query*** (Col. 21, Lines 43-48).

Regarding to claims 10 and 31, Fernandez teaches all the claimed subject matters as discussed in claims 1 and 22, Fernandez further discloses ***each tagger operator implements a next method to produce a result row*** (Col. 5, Line 57-Col. 9, Line 59).

Regarding to claims 11 and 32, Fernandez teaches all the claimed subject matters as discussed in claims 1 and 22, Fernandez further discloses the step of ***parsing said XML query; transforming said XML queries into a language-neutral intermediate representation; rewriting said language-neutral intermediate representation into an equivalent form easily translated into an SQL query; translating said equivalent form into one or more SQL queries over said relational database, and executing said one or more SQL queries to produce said results of said XML query over said relational database*** (Col. 5, Line 57-Col. 9, Line 59).

Regarding to claims 12 and 33, Fernandez teaches all the claimed subject matters as discussed in claims 11 and 32, Fernandez further discloses ***tagger graph is generated from said equivalent form*** (Col. 5, Line 57-Col. 9, Line 59).

Regarding to claims 13 and 34, Fernandez teaches all the claimed subject matters as discussed in claims 11 and 32, Fernandez further discloses ***tagger graph includes a tagger input operator for each node in a result XML tree of said XML query*** (Col. 5, Line 57-Col. 9, Line 59).

Regarding to claims 18 and 39, Fernandez teaches all the claimed subject matters as discussed in claims 11 and 32, Fernandez further discloses ***tagger operators comprise any of a tagger input operator, a tagger scalar operator or a tagger aggregate operator*** (Col. 20, Lines 44-49).

Regarding to claims 19 and 40, Fernandez teaches all the claimed subject matters as discussed in claims 11 and 32, Fernandez further discloses ***a number of relational database tables of said relational database are mapped to a number of virtual XML documents and said XML queries are issued over said virtual XML documents*** (Col. 3, Line 10-Col. 4, Line 61).

Regarding to claims 20 and 41, Fernandez teaches all the claimed subject matters as discussed in claims 1 and 22, Fernandez further discloses ***the method operates over a distributed computing network*** (FIG. 1).

Regarding to claims 21 and 42, Fernandez teaches all the claimed subject matters as discussed in claims 20 and 41, Fernandez further discloses ***the method operates over the Internet*** (Col. 4, Line 64-Col. 5, Line 10).

Regarding to claim 43, Fernandez teaches all the claimed subject matters as discussed in claim 22, Fernandez further discloses ***tagger runtime operates outside an RDBMS*** (Col. 5, Line 57-Col. 9, Line 59).

Allowable Subject Matter

4. Claims 2, 5-9, 14-17, 23, 26-30, 35-38 and 46 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

The following is an examiner's statement of reasons for allowance:

The closet available prior arts, USP 6,604,100, issued to Fernandez also teaches a method for tagging results of an XML query. However, as in claims 2, 5-9, 14-17, 23, 26-30, 35-38 and 46, Fernandez fails to teach or suggest ***tagger node graph has a top-most tagger operator and a plurality of lower-most tagger operators, said calling and evaluating steps further comprising: a. starting with said top-most tagger operator, each tagger operator implementing a method to request results from inputs to said tagger operator, said method causing lower-level tagger operators connected to said inputs to be called; b.***

starting with said lower-most tagger operators, each called tagger operator returning intermediate tagged results to a higher-level connected tagger operator upon evaluating said associated parse tree; performing steps a and b until an end of said results of said XML query is reached, and said top-most tagger operator producing tagged output XML of said results of said XML query as in claims 2, 23, 46, ***tagger input operators execute in a sorted outer union mode***, and ***said translating step produces a single SQL query to produce a single sorted outer union relational database result*** as in claims 5, 26 and 14, 35.

Conclusion

5. THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to HUNG Q PHAM whose telephone number is 703-605-4242. The examiner can normally be reached on Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, JOHN E BREENE can be reached on 703-305-9790. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Examiner Hung Pham
April 16, 2004

SHAHID ALAM
PRIMARY EXAMINER